WHAT IS CLAIMED IS:

1. In a server, a method of responding to a client request from a network connection, the network connection including a client-to-server channel and a server-to-client channel, the method comprising:

examining local server information to determine whether the client-to-server channel is still established;

inferring a state of the server-to-client channel according to whether the client-to-server channel is still established;

processing the client request if the inferred state indicates that the server-to-client channel is still established; and

terminating the client request if the inferred state indicates that the server-to-client channel is no longer established.

2. The method of claim 1, wherein the client request is read from the client-to-server channel; and wherein the state of the server-to-client channel is inferred after the client request has been read.

3. The method of claim 2, wherein the server includes a read buffer; wherein the request is read from the read buffer; and wherein the read buffer is then probed to determine whether the client-to-server channel is still established.

4. The method of claim 1, wherein the server maintains local information about the state of the connection; wherein a specific state of the connection is determined by examining the local information; wherein the client request is read and processed if the local information indicates that the connection is not in the specific state; and wherein the request is not processed if the local information indicates that the connection is in the specific state.

5. The method of claim 4, wherein the server-to-client channel is inferred to be no longer established if the local information indicates that the connection is in a "CLOSE_WAIT" state.

6. The method of claim 1, wherein the state of the server-to-client channel is inferred by polling the local information, the local information being polled while a response to the client request is being prepared, whereby a request can be terminated while the response is being prepared.

7. The method of claim 1, further comprising generating an interrupt when the server-to-client channel is inferred to be no longer established, wherein a response to the client request is processed until the interrupt is generated.

8. A network server comprising:
a processing unit;
a network interface card; and
computer memory encoded with an operating system including a

5   routine for commanding the processing unit to maintain a queue of connections based on connection requests received by the network interface card;

the computer memory being further encoded with a server program including a routine for commanding the processing unit to accept connections

10   from the queue; examine local server information to determine whether a client-to-server channel of a given connection from the queue is still established; process a client request associated with the given connection if the client-to-to-server channel of the given connection is still established; and terminate the associated client request if the client-to-server channel of the

15   given connection is no longer established.

9. The server of claim 8, wherein the client request is read from its associated client-to-server channel; and wherein a state of a server-to-client channel of the given connection is inferred after the client request has been read; the client request being processed or terminated according to the

5   inferred state.

10. The server of claim 9, further comprising a read buffer; wherein the client request is read into the read buffer; and wherein the read buffer is then probed to determine whether the client-to-server channel is still established.

11. The server of claim 8, wherein the memory includes local information about a state of the given connection; wherein a state of the given connection is determined by examining the local information; wherein the client request is read and processed if the local information indicates that the

5   given connection is not in a specific state; and wherein the request is not processed if the local information indicates that the given connection is in the specific state.

12. The server of claim 11, wherein a server-to-client channel of the given connection is inferred to be no longer established if the local information indicates that the given connection is in a "CLOSE_WAIT" state.

13. The server of claim 8, wherein a state of a server-to-client channel of the given connection is inferred by polling the local information, the local information being polled while a response to the client request is being prepared, the client request being processed or terminated according to the

5   inferred state, whereby a request can be terminated while the response is being prepared.

14. The server of claim 8, wherein the memory is further encoded with a routine for commanding the processing unit to generate an interrupt when a server-to-client channel of the given connection is inferred to be no longer established, and wherein a response to the client request is processed until
5   the interrupt is generated.

15. A network server comprising:

a processing unit;

first means for maintaining a queue of connections based on connection requests, each network connection including a client-to-server
5   channel and a server-to-client channel;

second means for accepting connections from the queue;

third means for examining local server information to determine whether the client-to-server channel of a given connection from the queue is still established;

10   fourth means for processing a client request associated with the given connection if it is determined that the client-to-server channel of the given connection is still established; and

fifth means for terminating the associated request if it is determined that the client-to-server channel of the given connection is no longer established.

16. An article of manufacture for a network server including a processing unit and a network interface card, the article comprising:

computer memory;

an operating system routine encoded in the computer memory, the
5   operating system routine, upon execution, commanding the processing unit to maintain a queue of connections based on connection requests received by the network interface card; and

a server program encoded in the computer memory, the server program including a routine for commanding the processing unit to accept connections
10   from the queue, each connection having a client-to-server channel and a

server-to-client channel; examine local server information to determine
whether the client-to-server channel of a given connection from the queue is
still established; process a client request associated with the given connection
if the client-to-server channel of the given connection is still established; and
15   terminate the associated request if the client-to-server channel of the given
connection is no longer established.

17. The article of claim 16, wherein the client request is read from the
client-to-server channel of the given connection; and wherein a state of the
server-to-client channel of the given connection is inferred after the client
request has been read.

18. The article of claim 16, wherein the memory is further encoded with
local information about a state of the given connection; wherein a state of the
given connection is determined by examining the local information; wherein
the client request is read and processed if the local information indicates that
5   the given connection is not in a specific state; and wherein the request is not
processed if the local information indicates that the given connection is in the
specific state.

19. The article of claim 16, wherein a state of the client-to-server
channel of the given connection is determined by polling the local information,
the local information being polled while a response to the client request is
being prepared, whereby a request can be terminated while the response is
5   being prepared.

20. The article of claim 16, wherein the memory is further encoded with
a routine for commanding the processing unit to generate an interrupt when
the client-to-server channel of the given connection is determined to be no
longer established, and wherein a response to the client request is processed
5   until the interrupt is generated.